

AMENDMENTS TO THE CLAIMS

The claims have been amended as follows:

Claims 1-7 (cancelled)

8. (Currently amended) A method for maintaining information about multiple instances of an activity related to a business process, comprising:

receiving process data regarding the instances from each of a plurality of application programs;

receiving continuation data regarding the instances, the continuation data correlating, for each of the instances, process data for the instance received from at least one of the application programs with process data for the same instance received from at least another of the application programs; and

inserting process data for each of the instances into instance database records based on the continuation data~~The method of claim 1,~~ wherein:

the instances are acted upon in a sequence of processing steps,

each of the applications provides process data corresponding to a different part of the processing sequence, and

process data for at least a portion of the instances are received in an order different from the processing sequence;~~and further comprising:~~

preventing access to instance database records containing out-of-order data reflecting completion of a processing step for an instance but not reflecting completion of a sequentially prior processing step for the instance.

9. (Previously presented) The method of claim 8, wherein said preventing access comprises preventing human users from viewing instance database records containing out-of-order data.

10. (Currently amended) The method of claim 8, wherein said preventing access comprises preventing one or more display or analysis application programs from performing display of or analysis upon records containing out-of-order data.

11. The method of claim 8, further comprising:

providing access to a first instance database record for an instance not containing out-of-order data; and

preventing access to a second instance database record for the instance, wherein the second instance database record contains out-of-order data, and wherein process data in the second instance database record is not correlated to process data in the first record by continuation data.

12. The method of claim 11, further comprising:

receiving correlation data indicating that the first and second records pertain to the same instance; and

merging the first and second records.

13. (Currently amended) The method of claim ~~18~~, wherein the process data is received in batch updates from the applications.

14. (Currently amended) The method of claim ~~13~~8, wherein:

~~the instances are acted upon in a sequence of processing steps,~~
~~each of the applications provides process data corresponding to a different part of the processing sequence,~~

~~process data for at least a portion of the instances are received in an order different from the processing sequence, and~~

process data from at least one of the applications is sequentially pre-sorted prior to batch update.

15 - 21. (cancelled)

22. (Currently amended) A computer-readable medium having stored thereon a program for maintaining information about multiple instances of an activity related to a business process which, when executed by a processor, cause the processor to perform steps comprising:

receiving process data regarding multiple instances of an activity from each of a plurality of application programs;

receiving continuation data regarding the instances, the continuation data correlating, for each of the instances, process data for the instance received from at least one of the application programs with process data for the same instance received from at least another of the application programs; and

inserting process data for each of the instances into instance database records based on the continuation data.~~The computer-readable medium of claim 15, wherein:~~

the instances are acted upon in a sequence of processing steps,

each of the applications provides process data corresponding to a different part of the processing sequence, and

process data for at least a portion of the instances are received in an order different from the processing sequence, and comprising additional data representing sequences of instructions which, when executed by a processor, cause the processor to perform additional steps comprising:

preventing access to instance database records containing out-of-order data reflecting completion of a processing step for an instance but not reflecting completion of a sequentially prior processing step for the instance.

23. The computer-readable medium of claim 22, wherein said preventing access comprises preventing human users from viewing instance database records containing out-of-order data.

24. (Currently Amended) The computer-readable medium of claim 22, wherein said preventing access comprises preventing one or more display or analysis application programs from performing display of or analysis upon records containing out-of-order data.

25. The computer-readable medium of claim 22, comprising additional data representing sequences of instructions which, when executed by a processor, cause the processor to perform additional steps comprising:

providing access to a first instance database record for an instance not containing out-of-order data; and

preventing access to a second instance database record for the instance, wherein the second instance database record contains out-of-order data, and wherein process data in the second instance database record is not correlated to process data in the first record by continuation data.

26. The computer-readable medium of claim 25, comprising additional data representing sequences of instructions which, when executed by a processor, cause the processor to perform additional steps comprising:

receiving correlation data indicating that the first and second records pertain to the same instance; and

merging the first and second records.

27. (Currently amended). The computer-readable medium of claim ~~4~~22, wherein the process data is received in batch updates from the applications.

28. (Currently amended) The computer-readable medium of claim ~~27~~22, wherein:

~~the instances are acted upon in a sequence of processing steps,~~

~~each of the applications provides process data corresponding to a different part of the processing sequence,~~

~~process data for at least a portion of the instances are received in an order different from the processing sequence, and~~

process data from at least one of the applications is sequentially pre-sorted prior to batch update.

29. (New) In a system having multiple application programs providing data with respect to activities related to particular instances of a business process to a process instance database, a computing device for managing the intake of and access to said data to and from said instance database, said computing device comprising:

- a first receiving unit to receive process data regarding the instances from each of a plurality of application programs;

- a second receiving unit to receive continuation data regarding the instances, the continuation data correlating, for each of the instances, process data for the instance received from at least one application program with process data for the same instance received from at least one other application program;

- an inserting unit to insert process data for each of the instances into instance database records based on the continuation data;

- a sequencing unit that tracks and manages the incoming application data for each instance so that it corresponds with a processing sequence that determines an order of steps defined by a process.

- an ordering unit that identifies process data for any instance that is received in an order different from the processing sequence;

- a limiting unit that prevents access to database records containing out-of-order data reflecting completion of a processing step for an instance but not reflecting completion of a sequentially prior processing step of the instance;

- a correlating unit that properly correlates database records of out-of-order processing data for an instance with the remaining processing data for that instance; and

- a merging unit that merges out-of-order processing data records for an instance with in-order processing data records for the same instance where the in-order and out-of-order data records are correlated by said correlating unit.

30. (New) A method for maintaining information about multiple instances of an activity related to a business process, comprising:

receiving process data regarding the instances from each of a plurality of application programs;

receiving continuation data regarding the instances, the continuation data correlating, for each of the instances, process data for the instance received from at least one of the application programs with process data for the same instance received from at least another of the application programs; and

inserting process data for each of the instances into instance database records based on the continuation data, wherein each instance database record comprises a primary key, a timestamp field for then the data was received, a field noting the geographical origin of the data, a field noting the size of the data, a field noting the time the data was collected, and field noting the time the data was transmitted, and further wherein:

the instances are acted upon in a sequence of processing steps,

each of the applications provides process data corresponding to a different part of the processing sequence, and

process data for at least a portion of the instances are received in an order different from the processing sequence; and

preventing access to instance database records containing out-of-order data reflecting completion of a processing step for an instance but not reflecting completion of a sequentially prior processing step for the instance wherein said out-of-order data records have not been correlated with the appropriate existing data records, said out-of-order records being similarly comprised to said instance database records and said out-of-order records having their data merged with existing records and then being deleted as part of said correlation process.